

Defining Fields

This section describes the field types, formats and types of constants available in Super Natural and how to obtain extended field descriptions from the Software AG productPredict.

This section covers the following topics:



What are Fields?



Field Types



Field Formats



Constants



Descriptors



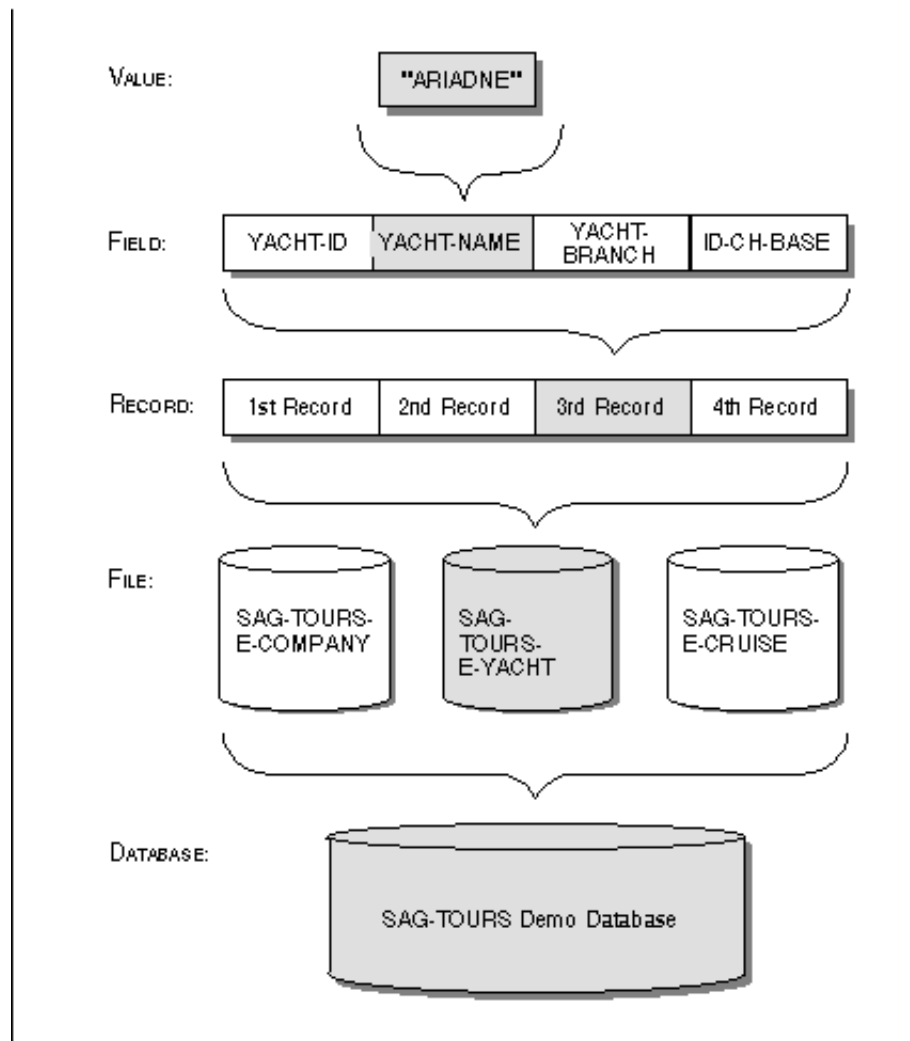
Extended Field Descriptions

What are Fields?

Fields are the smallest logical unit of information within a file. Fields are organized within records contained within files. For further information on records and files, see [Using a Database Structure and Files](#).

Super Natural identifies each field by a two-character reference which appears in the **Field Reference (Ref)** column on the Worksheet. Field types are indicated in the **Database Reference (DB)** column on the Worksheet. For further information, see [Working with the Worksheet](#).

The following diagram shows how the name of a yacht called Ariadne is stored in a database in the demonstration file **SAG-TOURS-E-YACHT**:



Field Types

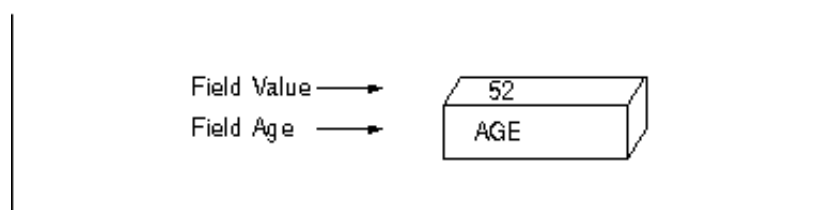
The following sections describe the field types available in Super Natural.

The examples are taken from an imaginary file containing personnel data.

Elementary Fields

Elementary fields contain one value per record, e.g., the field AGE contains only one age per person.

The following diagram shows an elementary field:

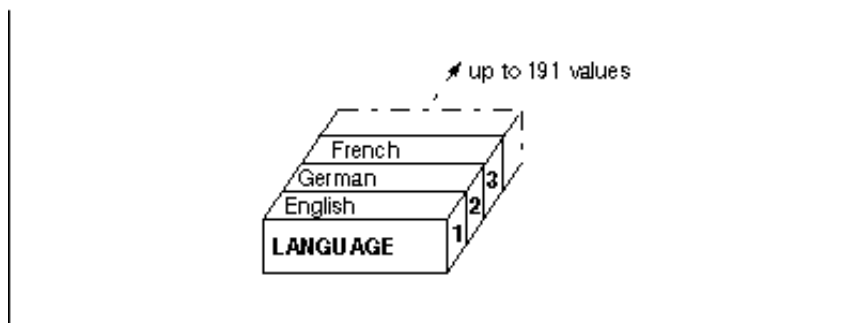


Multiple-Value Fields

Multiple-value fields can contain up to 191 values per record. For example, if a person speaks more than one language, you could use a multiple value field to store this information.

Each value within a multiple-value field is called an occurrence. You can deal with occurrences separately using referencing. For further information on referencing, see Referencing Occurrences in **Using Editors**.

The following diagram shows a multiple-value field which has three occurrences:



Groups

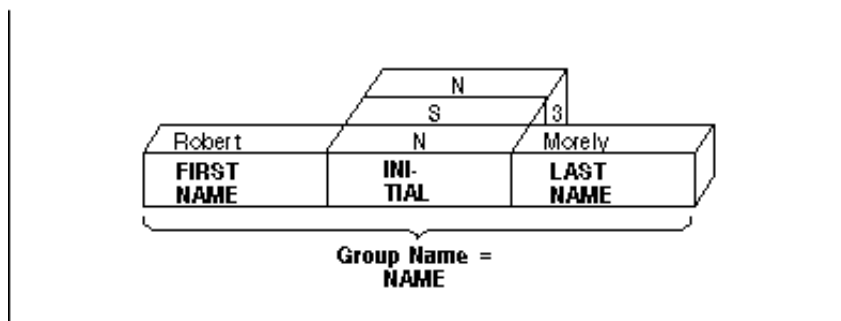
Groups connect related fields. For example, the group **NAME** connects the fields; **SURNAME**, **FIRST-NAME-1**, **FIRST-NAME-2**, **TITLE** and **FORM-OF-ADDRESS**.

A group can contain elementary fields and multiple-value fields. Groups can be contained or nested within other groups.

You can only use groups in the Sequence (Seq) column of the Worksheet. If you mark a group with a number in the Sequence (Seq) column, all the fields belonging to the group already listed in the Field Name column are automatically numbered.

If a group contains multiple-value fields or nested groups, you can limit the number of occurrences to be displayed in reports by modifying your Transaction Options. For further information on Transaction Options, see **Adjusting a User Profile**.

The following diagram shows a group which contains a multiple-value field:



Periodic Groups

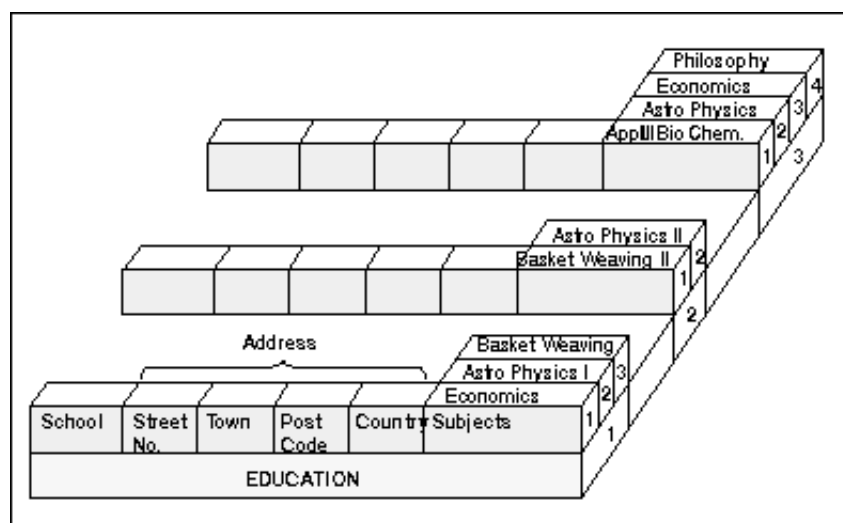
Periodic groups contain series of elementary and/or multiple-value fields which occurs more than once.

The values contained within a periodic group are structured like records. These records within records' are called occurrences. A periodic group can have up to 99 occurrences.

You can limit the number of occurrences to be displayed in reports by modifying your Transaction Options. For further information on Transaction Options, see [Adjusting a User Profile](#).

You can deal with occurrences separately using referencing. For further information on referencing, see [Referencing Occurrences in Using Editors](#).

The following diagram shows a periodic group:



User Fields

User fields are not contained in a database file. User fields are defined by the user for a particular transaction. User fields are most commonly used in calculations, logical condition processing and SQL. For information on how to define user fields, see [Field Name Column in Working with the Worksheet](#).

Dynamic Input Fields

Dynamic input fields are a kind of user field which you can use to execute a transaction several times using different values. You can use dynamic input fields in the same way as any other user field. Each time you run a transaction using dynamic input fields, Super Natural asks you to supply a value for each one. This means that you do not have to modify the transaction to see what happens when a new value is used.

You can use dynamic input fields in the Selection, Calculation and Logical Conditions editors.

Note:

You can not run transactions using dynamic input fields in batch mode.

System Variables

System variables show information which comes directly from the system.

You can obtain a selection list of system variables by entering two asterisks (**) in the Field Name column in the Worksheet.

System variables are described in detail in [section System Variables](#).

Input Fields

Input fields are fields from PC or Work files which you use with the Data Selection modes PC FILE and WORK FILE.

Field Formats

The field format determines which values can be assigned to a field and how long it can be. Each format is identified by a code.

Codes and Lengths

The following table gives an overview of the formats available in Super Natural.

Format	Code	Length in Characters (Bytes)
Alphanumeric	A	1 - 253
Binary	B	1 - 126
Date	D	No length specification
Floating Point	F	4 or 8
Integer	I	1, 2 or 4
Numeric	N	1-29
Packed Numeric	P	1-29
Time	T	No length specification

The following sections describe the values which can be assigned to fields in each format.

Alphanumeric Format

Fields of alphanumeric (A) format can contain the following values: the letters of the alphabet, special characters, numerical digits, blanks or any combination thereof. The following are examples of alphanumeric values:

Examples

```
06024/71-0 #504
Ernest Schneckenecker & Co.
BEAU300864TTRN20
```

Binary Format

Binary (B) fields can only contain the following values:

Format	Value
B1	0 - 255
B2	0 - 65535
B3	0 - 16777215
B4	-2147483648 - 2147483647
B5 - B126 Fields of these formats can contain any binary value.	

Fields of the formats B1, B2, B3 and B4 can also contain alphanumeric values.

Date Format

The date (D) format you use in Super Natural is dependent on the Natural parameter DTFORM setting at your site. The following formats are possible where Y=year, M=month and D=day:

DD/MM/YYYY	(European)
DD.MM.YYYY	(German)
YYYY-MM-DD	(International)
MM/DD/YYYY	(USA)

Floating Point Format

Floating point (F) format is useful for representing numeric values accurately which would otherwise be too long. The values in fields with floating point format consist of a mantissa (number) and an exponent, as in the following examples:

Example 1

+1.234E-10

Where 1.234 is the mantissa and -10 is the exponent. The value represented is 0.0000000001234.

Example 2

-12345678E+12

Where -12345678 is the mantissa and +12 is the exponent. The value represented is -12345678000000000000.

Integer Format

Fields of integer (I) format can only have the following values:

Format	Maximum Negative Value	Maximum Positive Value
I1	-128	+127
I2	-32768	+32767
I4	-2147483648	+2147483647

As integers are whole numbers, decimals are not allowed!

Numeric Format

Fields of numeric (N) format can contain the following values: numerals, the special characters "+", "-", and the decimal character. The character which is used as the decimal character depends on the Natural settings (DC parameter) at your site. The following are examples of numeric values:

Examples

```
1000
-15
+0.0258
```

Numeric values can have a total of 29 numerals (the decimal character and + and - signs are not included in the field length). You can have a maximum of seven numerals after the decimal character.

Note:

The term *numeric field* can refer to fields of the following formats: B, F, I, N and P.

Packed Numeric Format

Packed Numeric (P) values do not take up as much space internally as numeric values. Values in packed numeric format are represented in numeric format on your screen and in reports.

Values in packed numeric format can have a total of 29 numerals (the decimal character and + and - signs are not included in the field length). You can have a maximum of seven numerals after the decimal character.

Time Format

Values in time (T) format are presented as follows where H=hour, M=minute and S=second:

```
HH:MM:SS
```

Constants

Constants are values which cannot change - the opposite of variables. The following are examples of the use of constants:

-

Selection

```
NAME = 'SHAW'
```

-

Calculation

```
SALARY = SALARY * 1.2
```

-

Output

```
IF NAME = 'EMIL'
THEN USERFIELD = '*****'
ELSE USERFIELD = '_ '
```



Joining fields

RESULT-FIELD = NAME ! ' , ' ! FIRST-NAME



Assigning values to a field

HOBBY = 'NOT/APPLICABLE'

SAVINGS = 0

The following table shows how to represent the types of constants available in Super Natural:

Constant	Representation	Values
Alphanumeric	'value'	Value in format A. An apostrophe within an alphanumeric constant is represented by double apostrophes. Eg: 'HE SAID, ''HELLO'''
Date	D'value'	Value in format D as defined at your site
Extended Time	E'value'	Value is composed of the date in format D as defined at your site and the time in format T. Eg: E'1991-07-05 17:29:59'
Floating Point	value	Value in format F
Hexadecimal	H'value'	Value in format B. Value must be hexadecimal. For further information on hexadecimal values, see Glossary.
Integer	value	Value in format I
Numeric	value	Value in format N
Packed	value	Value in format P
Time	T'value'	Value in format T

Note:

Alphanumeric, date, extended time, time and hexadecimal constants must be enclosed within apostrophes ('.....'). Constants which can only contain numeric values must not be enclosed within apostrophes.

Descriptors

Descriptors enable Super Natural to search for data in the database without having to read every record. The database administrator can use both elementary and multiple-value fields to define descriptors.

The following sections describe the types of descriptors available in Super Natural.

Hyperdescriptors - ADABAS only

Hyperdescriptors are user exits in Adabas and function like subdescriptors and superdescriptors in Super Natural. Ask your administrator how to use hyperdescriptors at your site.

Key Field Descriptors

There are two types of descriptors known as key fields in Super Natural.

Elementary Key Fields

Elementary key fields are elementary fields which have been defined as descriptors.

Multiple-Value Key Fields

Multiple-value key fields are multiple-value fields which have been defined as descriptors.

Non-Database Descriptors - Adabas only

A non-database descriptor is an elementary or multiple-value field which can be used in selection criteria in the same way as a descriptor, but is not defined as such in the database.

Note:

When you use a non-database descriptor in the selection criteria, Adabas reads every record in the database.

Phonetic Descriptors - Adabas only

You can use phonetic descriptors to perform phonetic searches on fields. Phonetic searches result in the return of all values which sound similar to the phonetic descriptor you are using as a search value. Phonetic searches are useful if you are not sure how to spell what you are looking for! You can only use phonetic descriptors in the selection editor.

For further information on how to use phonetic descriptors, see [Using Editors](#).

Subdescriptors - Adabas only

Subdescriptors are created from a portion of a field.

Superdescriptors - Adabas only

Superdescriptors are created from up to five fields (or parts thereof) which may or may not be descriptors. Superdescriptors can be of alphanumeric format or, if all the fields included are numeric, of binary format.

Extended Field Descriptions

Super Natural provides access to extended field descriptions stored in the Software AG data dictionary Predict. (This feature is only available at sites where Predict is installed.)

You can access extended field information by entering **PI** in the **Information** column either of the **Worksheet** or of any **Field Selection List** window.

